(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number : **0 484 047 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number : **91309823.2**

(22) Date of filing : **23.10.91**

(51) Int. Cl.[5] : **G10H 1/00**, G10H 1/26

(30) Priority : **01.11.90 US 608111**

(43) Date of publication of application : **06.05.92 Bulletin 92/19**

(84) Designated Contracting States : **DE FR GB IT**

(71) Applicant : **International Business Machines Corporation
Old Orchard Road
Armonk, N.Y. 10504 (US)**

(72) Inventor : **Lisle, Ronald J.
1815 Beechnut Cove
Cedar Park, Texas 78613 (US)**
Inventor : **McDonald, B. Scott
115 Valley View Drive East
Leander, Texas 78641 (US)**
Inventor : **Wilkes, Michael D.
5832 Tributary Ridge Drive
Texas 78759 (US)**

(74) Representative : **Moss, Robert Douglas
IBM United Kingdom Limited Intellectual
Property Department Hursley Park
Winchester Hampshire SO21 2JN (GB)**

(54) Method and apparatus for simultaneous output of digital audio and midi synthesised music.

(57) A method and apparatus are disclosed for simultaneously outputting digital audio and MIDI synthesised music utilising a single digital signal processor. The Musical Instrumert Digital Interface (MIDI) permits music to be recorded and/or synthesised utilising a data file containing multiple serially listed program status messages and matching note on and note off messages. In contrast, digital audio is generally merely compressed, utilising a suitable data compression technique, and recorded. The audio content of such a digital recording may then be restored by decompressing the recorded data and converting that data utilising a digital-to-analog convertor. The method and apparatus of the present invention selectively and alternatively couples portions of a compressed digital audio file and a MIDI file to a single digital signal processor which alternately decompresses the digital audio file and implements a MIDI synthesiser. Decompressed audio and MIDI synthesised music are then alternately coupled to two separate buffers. The contents of these buffers are then additively mixed and coupled through a digital-to-analog convertor to an audio output device to create an output having concurrent digital audio and MIDI synthesised music.

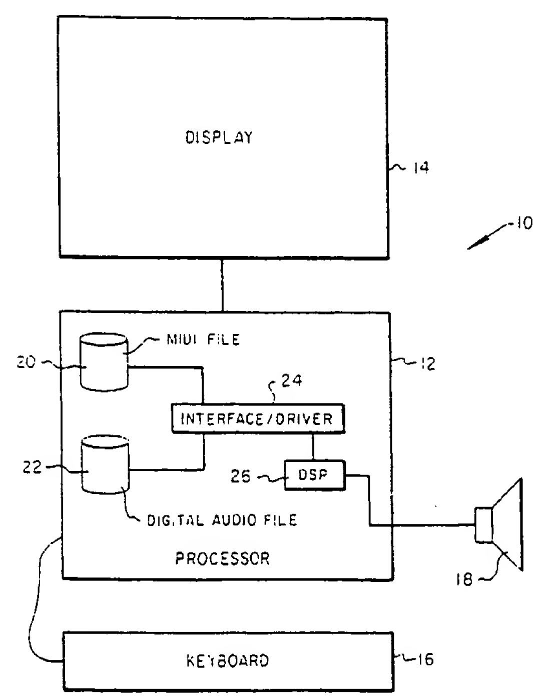

Fig. 1

EP 0 484 047 A2

Technical Field of the Invention

The present invention relates in general to the field of digital audio systems and in particular to systems which include MIDI synthesisers implemented utilising a digital signal processor. Still more particularly, the present invention relates to a method and apparatus for simultaneously outputting both digital audio and MIDI synthesised music utilising a single digital processor.

Background of the Invention

MIDI, the "Musical Instrument Digital Interface" was established as a hardware and software specification which would make it possible to exchange information such as: musical notes, program changes, expression control, etc. between different musical instruments or other devices such as: sequencers, computers, lighting controllers, mixers, etc. This ability to transmit and receive data was originally conceived for live performances, although subsequent developments have had enormous impact in recording studios, audio and video produuction, and composition environments.

A standard for the MIDI interface has been prepared and published as a joint effort betweeh the MIDI Manufacturer's Association (MMA) and the Japan MIDI Standards Committee (JMSC). This standard is subject to change by agreement between JMSC and MMA and is currently published as the MIDI 1.0 Detailed Specification, Document Version 4.1, January 1989.

The hardware portion of the MIDI interface operates at 31.25 KBaud, asynchronous, with a start bit, eight data bits and a stop bit. This makes a total of ten bits for a period of 320 microseconds per serial byte. The start bit is a logical zero and the stop bit is a logical one Bytes are transmitted by sending the least significant bit first. Data bits are transmitted in the MIDI interface by utilising a five milliamp current loop. A logical zero is represented by the current being turned on and a logical one is represented by the current being turned off. Rise times and fall times for this current loop shall be less than two microseconds. A five pin DIN connector is utilised to provide a connection for this current loop with only two pins being utilised to transmit the current loop signal. Typically, an opto-isolater is utilised to provide isolation between devices which are coupled together utilising a MIDI format.

Communication utilising the MIDI interface is achieved through multi-byte "messages" which consist of one status byte followed by one or two data bytes. There are certain exceptions to this rule. MIDI messages are sent over any of sixteen channels which may be utilised for a variety of performance information. There are five major types of MIDI messages: Channel Voice; Channel Mode; System Common; System Real-Time; and, System Exclusive. A MIDI event is transmitted as a message and consists of one or more bytes.

A channel message in the MIDI system utilises four bits in the status byte to address the message to one of sixteen MIDI channels and four bits to define the message. Channel messages are thereby intended for the receivers in a system whose channel umber matches the channel number encoded in the status byte. A instrument may receive a MIDI message on more than one channel. The channel in which it receives its main instructions, such as which program number to be on and what mode to be in, is often referred to as its "Basic Channel." There are two basic types of channel messages, a Voice message and a Mode message. A Voice message is utilised to control an instrument's voices and Voice messages are typically sent over voice channels. A Mode message is utilised to define the instrument's response to Voice messages, Mode messages are generally sent over the instrument's Basic Channel.

System messages within the MIDI system may include Common messages, Real-Time messages, and Exclusive messages. Common messages are intended for all receivers in a system regardless of the channel that receiver is associated with. Real-Time messages are utilised for synchronisation and are intended for all clock based units in a system. Real-Time messages contain status bytes only, and do not include data bytes. Real-Time messages may be sent at any time, even between bytes of a message which has a different status. Exclusive messages may contain any number of data bytes and can be terminated either by an end of exclusive or any other status byte, with the exception of Real-Time messages. An end of exclusive should always be sent at the end of a system exclusive message System exclusive messages always include a manfacturer's identification code. If a receiver does not recognise the identification code it will ignore the following data.

As those skilled in the art will appreciate upon reference to the foregoing, musical compositions may be encoded utilising the MIDI standard and stored and/or transmitted utilising substantially less data The MIDI standard permits the transmittal of a serial listing of program status messages and channel messages, such as "note on" and "note off" and as a consequence require substantially less digital data to encode than the straightforward digitisation of an analog music signal.

Earlier attempts at integrating music and other analog forms of communication, such as speech, into the digital computer area have traditionally involved the sampling of an analog signal at a sufficiently high frequency to ensure that the highest frequency present within the signal will be captured (the "Nyquist rate") and the subsequent digitisation of those samples for storage. The data rate required for such sim-

ple sampling systems can be quite enormous with several tens of thousands of bits of data being required for each second of audio signal.

As a consequence, many different encoding systems have been developed to decrease the amount of data required in such systems. For example, many modern digital audio systems utilise pulse code modulation (PCM) which employs a variation of a digital signal to represent analog information. Such systems may utilise pulse amplitude modulation (PAM), pulse duration modulation (PDM) or pulse position modulation (PPM) to represent variations in an analog signal.

One variation of pulse code modulation, Delta Pulse Code Modulation (DPCM) achieves still further data compression by encoding only the difference between one sample and the next sample. Thus, despite the fact that an analog signal may have a substantial dynamic range, if the sampling rate is sufficiently high so that adjacent signals do not differ greatly, encoding only the difference between two adjacent signals can save substantial data. Further, adaptive or predictive techniques are often utilised to further decrease the amount of data necessary to represent an analog signal by attempting to predict the value of a signal based upon a weighted sum of previous signals or by some similar algorithm.

In each of these digital audio techniques speech or an audio signal may be sampled and digitised utilising straightforward processing and digital-to-analog or analog-to-digital conversion techniques to store or recreate the signal.

While the aforementioned digital audio systems may be utilised to accurately store speech or other audio signal samples a substantial penalty in data rates must be paid in order to achieve accurate results over that which may be achieved in the music world with the MIDI system described above. However, in systems wherein it is desired to recreate human speech there exists no appropriate alternative in the MIDI system for the reproduction of human speech.

Disclosure of the Invention

Thus, a need exists for a method and apparatus whereby certain digitised audio samples, such as human speech, may be recreated and combined with synthesised music which was created or recreated utilising a MIDI data file.

The invention provides, in one aspect, a method for the simultaneous output of digital audio and MIDI synthesised music by a single digital signal processor, said method comprising the, steps of: storing a compressed digital audio file within a memory device associated with a single digital signal processor; storting a MIDI file within a memory device associated with said single digital signal processor; selectively and alternatively coupling portions of said compressed digital audio file to said single digital signal processor for creation of decompressed audio and portions of said MIDI file to said single digital signal processor for creation of MIDI synthesised music; storing said decompressed digital audio within a first temporary buffer; storing said MIDI synthesised music within a second temporary buffer; and combining the contents of said first temporary buffer and said second temporary buffer to create a composite output including digital audio and MIDI synthesised music.

In a second aspect, the invention provides apparatus for simultaneously outputting digital audio and MIDI synthesised music, said apparatus comprising: first memory means for storing a compressed digital audio file; second memory means for storing a MIDI file; a single digital signal processor; control means for selectively and alternatively coupling said first memory means to said single digital signal processor for creation of decompressed audio, and said second memory means to said single digital signal processor for creation of MIDI synthesised music; first buffer means coupled to said single digital signal processor for temporarily storing decompressed audio; second buffer means coupled to said single digital signal processor for temporarily storing MIDI synthesised music; and additive mixer means coupled to said first buffer means and said second buffer means for creating a composite output including digital audio and MIDI synthesised music.

Thus the invention provides an improved method and apparatus for simultaneously outputting both digital audio and MIDI synthesised music utilising a single digital processor.

The Musical Instrument Digital Interface (MIDI) permits music to be recorded and/or synthesised utilising a data file containing multiple serially listed program status messages and matching note on and note off messages. In contrast, digital audio is generally merely compressed, utilising a suitable data compression technique, and recorded. The audio content of such a digital recording may then be restored by decompressing the recorded data and converting that data utilising a digital-to-analog convertor. The method and apparatus of the present invention selectively and alternatively couples portions of a compressed digital audio file and a MIDI file to a single digital signal processor which alternately decompresses the digital audio file and implements a MIDI synthesiser. Decompressed audio and MIDI synthesised music are then alternately coupled to two separate buffers. The contents of these buffers are then additively mixed and coupled through a digital-to-analog convertor to an audio output device to create an output having concurrent digital audio and MIDI synthesised music.

A preferred embodiment of the invention will now be described, by way of example only, with reference to the accompanying drawings:

Brief Description of the Drawings

Figure 1 is a block diagram of a computer system which may be utilised to implement the method and apparatus of the present invention;

Figure 2 is a block diagram of an audio adapter which includes a digital signal processor which may be utilised to implement the method and apparatus of the present invention; and

Figure 3 is a high level flow chart and timing diagram of the method and apparatus of the present invention.

Detailed Description of the Invention

With reference now to the figures and in particular with reference to **Figure 1**, there is depicted a block diagram a computer system **10** which may be utilised to implement the method and apparatus of the present invention. As is illustrated, a computer system **10** is depicted. Computer system **10** may be implemented utilishng any state-of-the-art digital computer system having a suitable digital signal processor disposed therein which is capable of implementing a MIDI synthesiser. For example, computer system **10** may be implemented utilising an IBM PS/2 type computer which includes an IBM Audio Capture & Playback Adapter (ACPA).

Also included within computer system **10** is display **14**. Display **14** may be utilised, as those skilled in the art will appreciate, to display those command and control features typically utilised in the processing of audio signals within a digital computer system. Also coupled to computer system **10** is computer keyboard **16** which may be utilised to enter data and select various files stored within computer system **10** in a manner well known in the art. Of course, those skilled in the art will appreciate that a graphical pointing device, such as a mouse or light pen, may also be utilised to enter commands or select appropriate files within computer system **10**.

Still referring to computer system **10**, it may be seen that processor **12** is depicted. Processor **12** is preferably the central processing unit for computer system **10** and, in the depicted embodiment of the present invention, preferably includes an audio adapter capable of implementing a MIDI synthesiser by utilising a digital signal processor. One example of such a device is the IBM Audio Capture & Flayback Adapter (ACPA).

As is illustrated, MIDI file **20** and digital audio file **22** are both depicted as stored within memory within processor **12**. The output of each file may then be coupled to interface/driver circuitry **24**. Interface/driver circuitry **24** is preferably implemented utilising any suitable audio application programming interface which permits the accessing of MIDI protocol files or digital audio files and the coupling of those files to an appropriate device driver circuit within interface/driver circuitry **24**.

Thereafter, the output of interface/driver circuitry **24** is coupled to digital signal processor **26**. Digital signal processor **26**, in a manner which will be explained in greater detail herein, is utilised to simultaneously output digital audio and MIDI synthesised music and to couple that output to audio output device **18**. Audio output device **18** is preferably an audio speaker or pair of speakers in the case of stereo music files.

Referring now to **Figure 2**, there is depicted a block diagram of an audio adapter which includes digital signal processor **26** which may be utilised to implement the method and apparatus of the present invention. As discussed above, this audio adapter may be simply implemented utilising the IBM Audio Capture & Playback Adapter (ACPA) which is commercially available. In such an implementation digital signal processor **26** is provided by utilising a Texas Instruments TMS 320C25, or other suitable digital signal processor.

As illustrated, the interface between processor **12** and digital signal processor **26** is I/O bus **30**. Those skilled in the art will appreciate that I/O bus **30** may be implemented utilising the Micro Channel or PC I/O bus which are readily available and understood by those skilled in the personal computer art. Utilising I/O bus **30**, processor **12** can access the host command register **32**. Host command register **32** and host status register **34** are used by processor **12** to issue commands and monitor the status of the audio adapter depicted within **Figure 2**.

Processor **12** may also utilise I/O bus **30** to access the address high byte latched counter and address low byte latched counter which are utilised by processor **12** to access shared memory **48** within the audio adapter depicted within **Figure 2**. Shared memory **48** is preferably an 8K x 16 fast static RAM which is "shared" in the sense that both processor **12** and digital signal processor **26** may access that memory. As will be discussed in greater detail herein, a memory arbiter circuit is utilised to prevent processor **12** and digital signal processor **26** from accessing shared memory **48** simultaneously.

As is illustrated, digital signal processor **26** also preferably includes digital signal processor control register **36** and digital signal processor status register **38** which are utilised, in the same manner as host command register **32** and host status register **34**, to permit digital signal processor **26** to issue commands and monitor the status of various devices within the audio adapter.

Processor **12** may also be utilised to couple data to and from shared memory **48** via I/O bus **30** by utilising data high byte bi-directional latch **44** and data low-byte bi-directional latch **46**, in a manner well known in the art.

Sample memory **50** is also depicted within the

audio adapter of Figure **2**. Sample memory **50** is preferably a 2K x 16 static RAM which is utilised by digital signal processor **26** for outgoing samples to be played and incoming samples of digitised audio. Sample memory **50** may be utilised, as will be explained in greater detail herein, as a temporary buffer to store decompressed digital audio samples and MIDI synthesised music samples for simultaneous output in accordance with the method and apparatus of the present invention. Those skilled in the art will appreciate that by decompressing digital audio data and by creating synthesised music from MIDI files unit a predetermined amount of each data type is stored within sample memory **50**, it will be a simple matter to combine these two outputs in the manner described herein.

Control logic **56** is also depicted within the audio adapter of **Figure 2.** Control logic **56** is preferably a block of logic which, among other tasks, issues interrupts to processor **12** after a digital signal processor **26** interrupt request, controls the input selection switch and issues read, write and enable strobes to the various latches and memory devices within the audio adapter depicted. Control logic **56** preferably accomplishes these tasks utilising control bus **58**.

Address bus **60** is depicted and is preferably utilised, in the illustrated embodiment of the present invention, to permit addresses of various samples and files within the system to be coupled between appropriate devices in the system. Data bus **62** is also illustrated and is utilised to couple data among the various devices within the audio adapter depicted.

As discussed above, control logic **56** ,also uses memory arbiter logic **64** and **66** to control access to shared memory **48** and sample memory **50** to ensure that processor **12** and digital signal processor **26** do not attempt to access either memory simultaneously. This technique is well known in the art and is necessary to ensure that memory deadlock or other such symptoms do not occur.

Finally, digital-to-analog converter **52** is illustrated and is utilised to convert the decompressed digital audio or digital MIDI synthesised music signals to an appropriate analog signal. The output of digital-to-analog converter **52** is then coupled to analog output section **68** which, preferably includes suitable filtration and amplification circuitry. Similarly, the audio adapter depicted within **Figure 2** may be utilised to digitise and store audio signals by coupling those signals into analog input section **70** and thereafter to analog-to-digital converter **54**. Those skilled in the art will appreciate that such a device permits the capture and storing of analog audio signals by digitisation and storing of the digital values associated with that signal.

With reference now to **Figure 3**, there is depicted a high level flow chart and timing diagram of the method and apparatus of the present invention. As illustrated, the process begins at block **100** which depicts the retrieving of a compressed digital audio data block from memory. Thereafter, in the sequence depicted numerically, the digital audio data is decompressed utilising digital signal processor **26** and an appropriate decompression technique. Those skilled in the art will appreciate that the decompression technique utilised will vary in accordance with the compression technique which was utilised. Next, the decompressed digital audio data is loaded into a temporary buffer, such as sample memory **50** (see **Figure 2**).

At this point, in accordance with an important feature of the present invention, digital signal processor **26** is selectively and alternatively utilised to implement a MIDI synthesiser. This process begins at block **106** which depicts the retrieval of MIDI data from memory. Next, block **108** illustrates the creation of synthesised music by coupling the various program status changes, note on and note off messages and other control messages within the MIDI data file to a digital synthesiser which may be implemented utilising digital signal processor **26**. Thereafter, the synthesised music created from that portion of the MIDI file which has been retrieved is also loaded into a temporary buffer, such as sample memory **50**.

At this point, the decompressed digital audio data and the synthesised music, each having been located into a temporary buffer, are combined in an additive mixer which serves to mix the digital audio data and synthesised music so that they may be, simultaneously output. The output of this additive mixer is then coupled to an appropriate digital-to-analog conversion device, as illustrated in block **114**. Finally, the output of the digital-to-analog conversion device is coupled to an audio output device, as depicted in block **116**.

Of course, those skilled in the art will appreciate that the illustrated embodiment is representative in nature and not meant to be all inclusive. For example, the system may be implemented with alternate timing in that MIDI data may be retrieved first followed by compressed digital audio data. Similarly, in the event eight note polyphony is desired, sufficient MIDI data must be retrieved from memory to synthesise each note which is active for the portion of synthesised music to be created. Similarly, in the event stereo music is created, various control signals such as a pan signal must also be included to ensure that the audio outputs are coupled to an appropriate speaker, with the desired amount of amplification in that channel.

Upon reference to the foregoing those skilled in the art will appreciate that the Applicants in the present application have developed a technique whereby compressed digital audio data may be decompressed and portions of that data stored within a temporary buffer while MIDI data files are accessed and utilised

to create digital synthesised music in a MIDI synthesiser which is implemented utilising the same digital signal processor which is utilised to decompress the digital audio data. By selectively and alternatively accessing these two diverse types of data and then additively mixing the two outputs, a single digital signal processor may be utilised to simultaneously output both decompressed digital audio data and MIDI synthesised music in a manner which was not heretofore possible.

Claims

1. A method for the simultaneous output of digital audio and MIDI synthesised music by a single digital signal processor, said method comprising the steps of:
   storing a compressed digital audio file (22) within a memory device associated with a single digital signal processor;
   storing a MIDI file (20) within a memory device associated with said single digital signal processor;
   selectively and alternatively coupling portions of said compressed digital audio file to said single digital signal processor for creation of decompressed audio, and portions of said MIDI file to said single digital signal processor for creation of MIDI synthesised music;
   storing said decompressed digital audio within a first temporary buffer;
   storing said MIDI synthesised music within a second temporary buffer; and
   combining the contents of said first temporary buffer and said second temporary buffer to create a composite output including digital audio and MIDI synthesised music.

2. A method as claimed in claim 1, further including the step of coupling said composite output to a digital-to-analog converter.

3. A method as claimed in claim 2, further including the step of coupling an output of said digital-to-analog converter to an audio output device.

4. A method as claimed in any preceding claim, wherein said step of selectively and alternatively coupling portions of said compressed digital audio file to said single digital signal processor for creation of decompressed audio and portions of said MIDI file to said single digital signal processor for creation of MIDI synthesised music comprises the step of coupling a selected portion of said compressed digital audio file to said single digital signal processor until a predetermined amount of decompressed audio is created.

5. A method as claimed in any preceding claim, wherein said step of selectively and alternatively coupling portions of said compressed digital audio file to said single digital signal processor for creation of decompressed audio and portions of said MIDI file to said single digital signal processor for creation of MIDI synthesised music comprises the step of coupling a selected portion of said MIDI file to said single digital signal processor until a predetermined amount of digitally synthesised music is created.

6. Apparatus for simultaneously outputting digital audio and MIDI synthesised music, said apparatus comprising:
   first memory means (48) for storing a compressed digital audio file (22);
   second memory means (48) for storing a MIDI file (20);
   a single digital signal processor (26);
   control means (56) for selectively and alternatively coupling said first memory means to said single digital signal processor for creation of decompressed audio, and said second memory means to said single digital signal processor for creation of MIDI synthesised music;
   first buffer means (50) coupled to said single digital signal processor for temporarily storing said decompressed audio;
   second buffer means (50) coupled to said single digital signal processor for temporarily storing MIDI synthesised music; and
   additive mixer means coupled to said first buffer means and said second buffer means for creating a composite output including digital audio and MIDI synthesised music.

7. The apparatus as claimed in Claim 6, further including a digital-to-analog converter (52) coupled to said additive mixer means for converting said composite output to an analog signal.

8. Apparatus as claimed in Claim 7, further including audio output means (68) coupled to said digital-to-analog converter for outputting said analog signal.

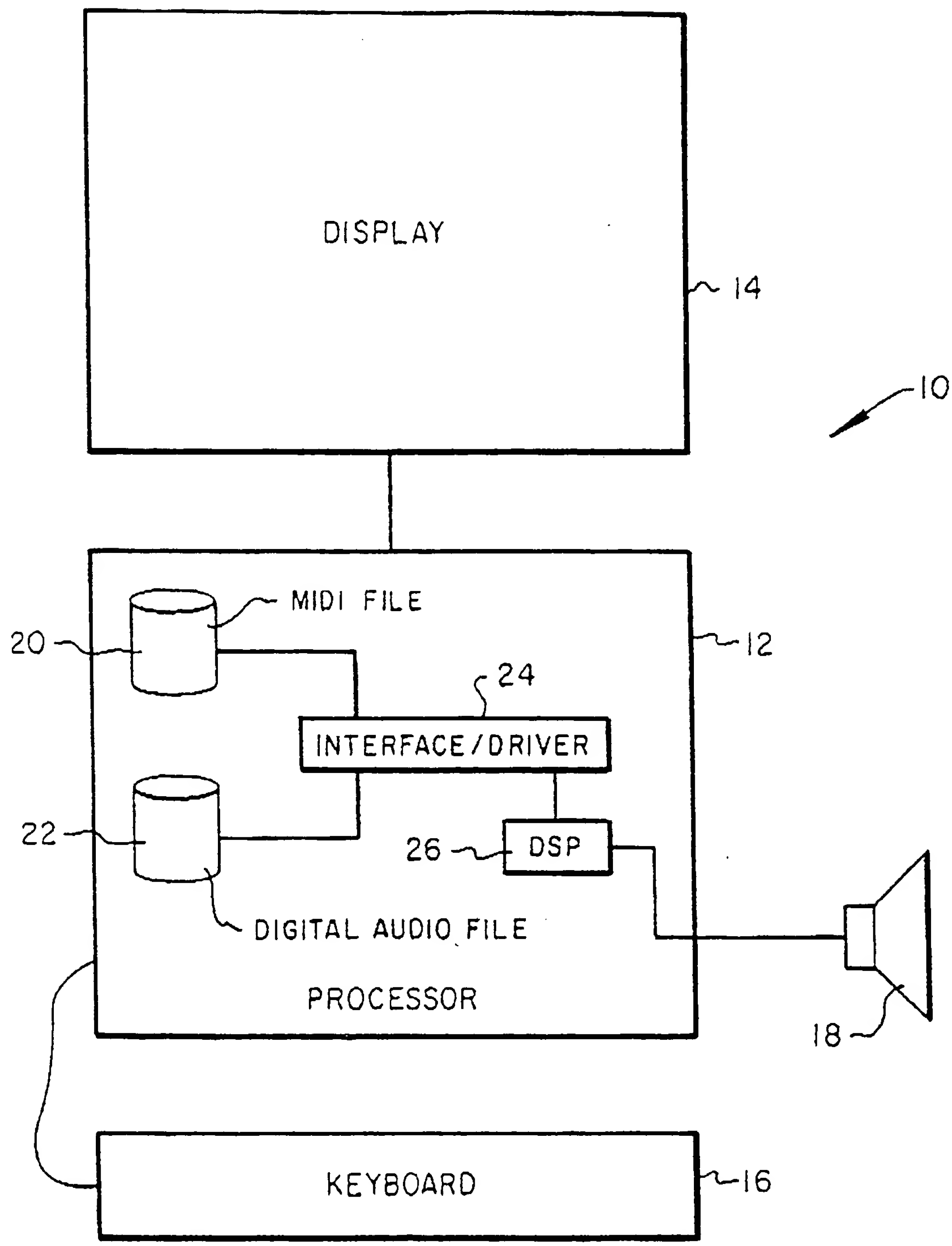
DISPLAY
14
10
MIDI FILE
20
12
24
INTERFACE/DRIVER
22
26
DSP
DIGITAL AUDIO FILE
PROCESSOR
18
KEYBOARD
16

Fig. 1

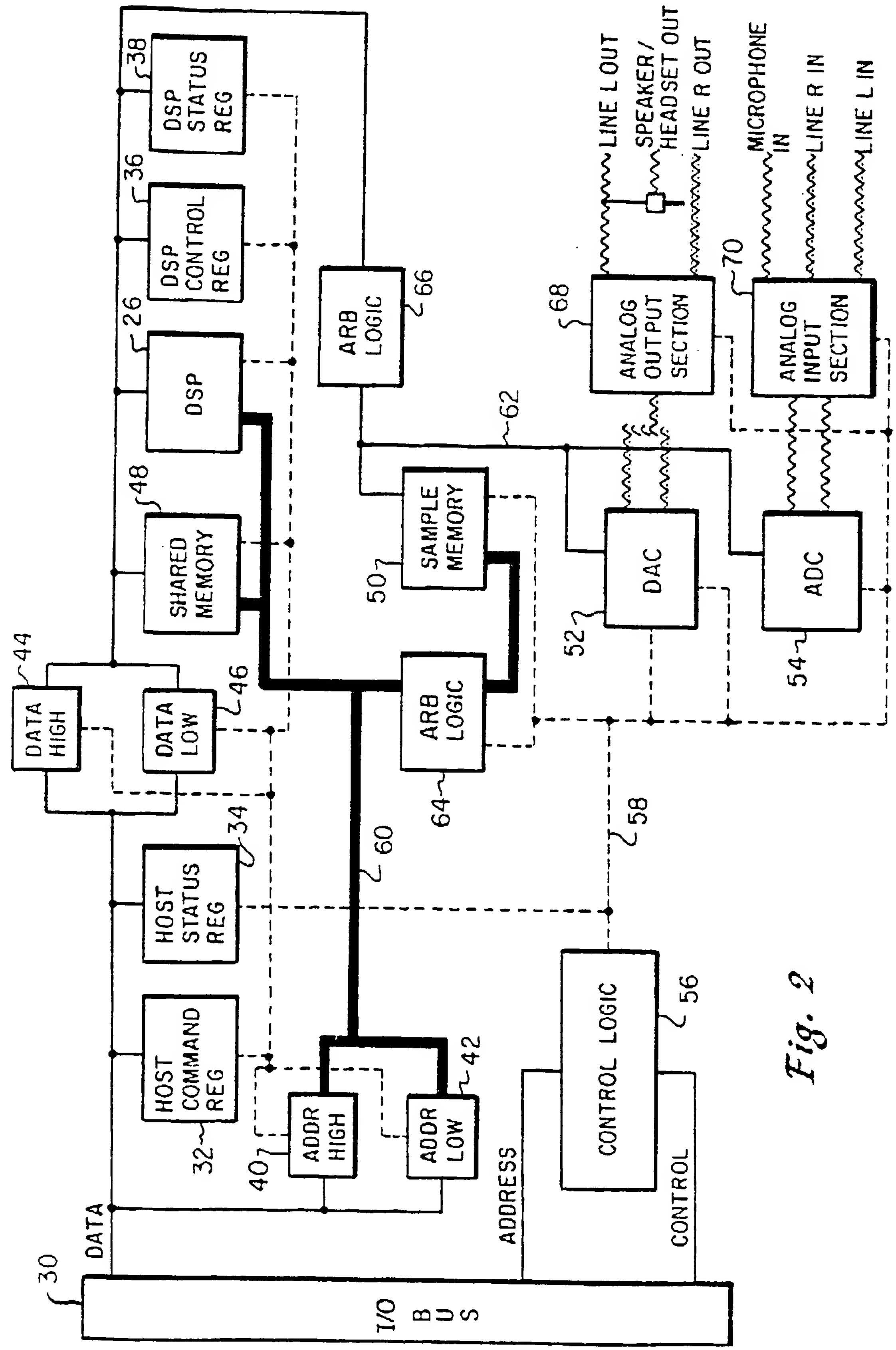
I/O BUS
DATA
ADDRESS
CONTROL
CONTROL LOGIC
HOST COMMAND REG
HOST STATUS REG
DATA HIGH
DATA LOW
ADDR HIGH
ADDR LOW
SHARED MEMORY
DSP
DSP CONTROL REG
DSP STATUS REG
ARB LOGIC
SAMPLE MEMORY
DAC
ADC
ANALOG OUTPUT SECTION
ANALOG INPUT SECTION
LINE L OUT
SPEAKER/ HEADSET OUT
LINE R OUT
MICROPHONE IN
LINE R IN
LINE L IN
30
32
34
36
38
40
42
44
46
48
50
52
54
56
58
60
62
64
66
68
70
26

Fig. 2

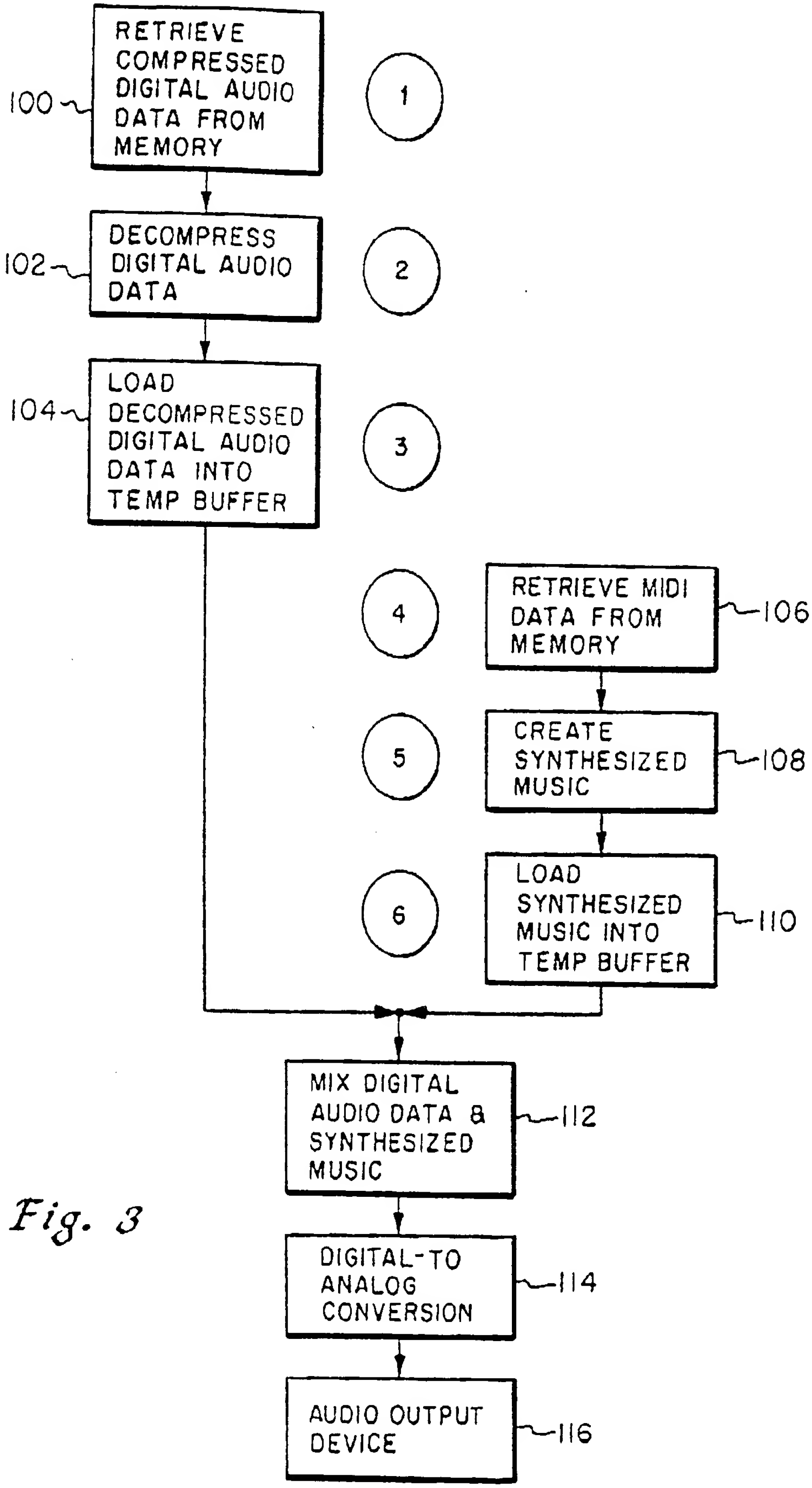
RETRIEVE COMPRESSED DIGITAL AUDIO DATA FROM MEMORY
100
1
DECOMPRESS DIGITAL AUDIO DATA
102
2
LOAD DECOMPRESSED DIGITAL AUDIO DATA INTO TEMP BUFFER
104
3
4
RETRIEVE MIDI DATA FROM MEMORY
106
5
CREATE SYNTHESIZED MUSIC
108
6
LOAD SYNTHESIZED MUSIC INTO TEMP BUFFER
110
MIX DIGITAL AUDIO DATA & SYNTHESIZED MUSIC
112
DIGITAL-TO ANALOG CONVERSION
114
AUDIO OUTPUT DEVICE
116

Fig. 3